





A3

**OPTIMUM LAYOUT STRUCTURE DETERMINATION METHOD FOR STRUCTURED DOCUMENT AND DETERMINATION METHOD FOR OPTIMUM LAYOUT TREE**

**Patent number:** JP7182340  
**Publication date:** 1995-07-21  
**Inventor:** MANTHA SURYANARAYANA M; BROWN JR ALLEN L;  
WAKAYAMA TOSHIRO  
**Applicant:** XEROX CORP  
**Classification:**  
- **international:** G06F17/24; G06K9/20  
- **european:**  
**Application number:** JP19940249622 19941014  
**Priority number(s):**

**Also published as:**

 EP0650130 (A2)  
 US5438512 (A1)  
 EP0650130 (A3)  
 EP0650130 (B1)

**Abstract of JP7182340**

**PURPOSE:** To provide specifications of the layout processing of a structured document in a computer-base document handling system.

**CONSTITUTION:** This method determines an optimum layout tree based on a logic tree and a set of grammar and includes a step where the logic tree is determined based on logical structure grammar by which logical structure information is separated from processing information, a step where the logic tree is parsed on the basis of grammar preliminarily determined to generate plural layout structures, a step where plural layout structures are evaluated based on weighting value calculation of selected attributes and weighting values are based on priority specifications, and a step where the optimum layout tree is determined based on evaluation of a priority specification base.

---

Data supplied from the **esp@cenet** database - Worldwide

**THIS PAGE BLANK (USPTO)**

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-182340

(43) 公開日 平成7年(1995)7月21日

(51) Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 17/24				
G 0 6 K 9/20	3 4 0 C	9288-5L	G 0 6 F 15/ 20	5 3 4 P

審査請求 未請求 請求項の数2 O L (全 12 頁)

(21) 出願番号 特願平6-249622

(22) 出願日 平成6年(1994)10月14日

(31) 優先権主張番号 1 3 9 6 8 6

(32) 優先日 1993年10月22日

(33) 優先権主張国 米国 (U S)

(71) 出願人 590000798

ゼロックス コーポレイション

XEROX CORPORATION

アメリカ合衆国 ニューヨーク州 14644

ロチェスター ゼロックス スクエア

(番地なし)

(72) 発明者 スルヤナラヤーナ エム. マンサ

アメリカ合衆国 ニューヨーク州 14580

ウェ ブスター カントリー マナー

ウェイ 100 アpartment ナンバー  
4

(74) 代理人 弁理士 中島 淳 (外2名)

最終頁に続く

(54) 【発明の名称】 構造化文書の最適レイアウト構造決定方法及び最適レイアウトツリーの決定方法

(57) 【要約】

【目的】 コンピュータベースのドキュメントハンドリングシステムでの構造化文書のレイアウト処理の仕様を提供する。

【構成】 論理ツリー及び文法の集合に基づいて最適レイアウトツリーを決定する方法であって、論理構造情報が処理情報と分離される論理構造文法に基づいて論理ツリーを決定するステップと、複数のレイアウト構造を生成するために予め定められた文法に基づいて前記論理ツリーを構文解析するステップと、選択された属性の重み付け値計算に基づいて前記複数のレイアウト構造を評価し、前記重み付け値が優先仕様にに基づくステップと、前記優先仕様ベースの評価に基づいて最適レイアウトツリーを決定するステップと、を含む。

1

## 【特許請求の範囲】

【請求項 1】 構造化文書の汎用論理構造が与えられると、構造化文書の最適レイアウト構造を決定する、有限状態マシンで実現される方法であって、関係属性文法で前記構造化文書の前記汎用論理構造を指定し、前記関係属性が二項関係で指定されるステップと、関係属性文法で構造化文書のクラスのための汎用レイアウト構造を指定するステップと、前記構造化文書の前記汎用論理構造と前記構造化文書の 10 前記汎用レイアウト構造のためにコーディネーション文法を決定し、関係属性文法で前記コーディネーション文法を表現するステップと、出力レイアウト構造中の潜在的な曖昧さを取り除くために優先仕様を決定するステップと、前記レイアウト構造を前記コーディネーション文法に基づいて前記汎用論理構造にリンクするステップと、前記優先仕様を用いてリンクされたレイアウト構造に前記論理構造を変換し、前記リンクステップ中に生成された曖昧さを取り除くステップと、 20 を含む構造化文書の最適レイアウト構造決定方法。

【請求項 2】 論理ツリー及び文法の集合に基づいて最適レイアウトツリーを決定する方法であって、論理構造情報が処理情報と分離される論理構造文法に基づいて論理ツリーを決定するステップと、複数のレイアウト構造を生成するために予め定められた文法に基づいて前記論理ツリーを構文解析するステップと、選択された属性の重み付け値計算に基づいて前記複数のレイアウト構造を評価し、前記重み付け値が優先仕様に 30 基づくステップと、前記優先仕様ベースの評価に基づいて最適レイアウトツリーを決定するステップと、を含む最適レイアウトツリーの決定方法。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明はコンピュータベースのドキュメントハンドリングシステム（文書処理システム）での構造化文書のレイアウト処理の仕様に關する。特に、本発明は、関係属性文法 (relational attributed grammar) の間で関係属性文法及びコーディネーション文法（調和文法）(coordination grammar) を用いて構造化文書のレイアウトを指定するための方法及び装置を提供することを目的とする。加えて、本発明は論理構造及びコーディネーション文法から得られるレイアウト構造中の潜在的な曖昧さを除くために優先的な制約条件 (preference 40

Article	→	HeaderBody
Header	→	TitleAuthors
Title	→	CHAR*
Authors	→	CHAR*

50

2

\*rential constrains) を指定し、部分的に最適化された解 (sub-optimal solutions) を除去するためにこれらの優先的な制約条件を用いるための方法及び装置を提供することを目的とする。

## 【0002】

【従来の技術】 コンピュータベースのドキュメントシステムは一般的に、テキストエディタ (text editor) 並びにワードプロセッサシステム、フォーマッタ、構文主導型エディタ (syntax directed editor)、及び特殊ツールの 4 つの広範なカテゴリに分けられる。殆どのシステムはこれらの 4 つの広範なカテゴリの内の 1 つよりも多い特徴を有する。プレゼンテーション的な考慮（要件）が重要である文書の場合、文書はプレゼンテーション前に準備のためにフォーマッタに提出されなければならない。

【0003】 フォーマッタはディスプレイ独立型レイアウト仕様又は装置依存型のレイアウト仕様のいずれかを生成するように文書処理する非対話型ツールである。文書は、ファイル記述形式をとってフォーマッタに提出され、ある一定時間後、フォーマッタは処理を完了し、全結果をリターンする。高水準フォーマッタは文書の論理的記述に基づき動作する。ユーザは、所望されるプレゼンテーションの詳細を指定することを要求されない。ユーザは文書の論理体系、即ち例えば、セクション、パラグラフ、ヘッダ、サマリー等、文書中に見られる異なるタイプの要素を処理する。フォーマッタはこれらの要素のレイアウトプレゼンテーションを処理する。低水準フォーマッタは、例えばフォント、スペース、マージン、ジャスティフィケーション（行揃え）等の文書の他の特性の変更を可能にするように文書の記述内にコマンドを含むことを可能にする。本発明は主に高水準フォーマッタを提供することを目的とする。

【0004】 文書のレイアウトが用意されている時に、たいいていの対話型システム (interactive system) ではユーザが文書のレイアウトを見ることができる。これらの対話型システムは、プレゼンテーション細部の仕様から文書の論理構造を分離する。一般的に高水準フォーマッタと同様に対話型システムは、文書の論理構造を記述するために文法的な記法を用いる。これらの論理構造は、事実上、主として階層的であり、論理構造を表現するためにツリー構造が使用される。図 1 は単純なツリー構造の実例である。技術的なアーティクル（論文）(article) の構造は、例えば以下の文法規則の集合即ちプロダクションとして表現され得る。

## 【0005】

3		
Body	→	Section*
Section	→	SectionTitle Paragraph* Section
Title	→	STRING CHAR*
Paragraph	→	Entity*
Entity	→	Text   TABLEEntity   List
Text	→	CHAR*
TableEntity	→	Table Caption
Caption	→	STRING CHAR*
List	→	Item*
Item	→	ItemMark Paragraph*
Item Mark	→	STRING

4

【0006】プロダクションの上記集合において、例えばCHAR、TABLE及びSTRING等の大文字で見られるワードは、終端シンボル(terminal symbol)である。終端シンボルはそれ以上の内部構造を有さない。残りのシンボルは非終端シンボル(non-terminal symbol)である。シンボルは要素タイプとも呼ばれる。プロダクションは左側の非終端シンボルの構造を特定する。例えば、第1のプロダクションに従って、Article(論文)は、Body(ボディ)がHeader(ヘッダ)の次に続いて構成される。演算子“\*”は、その前に0個以上のシンボルの発生を意味する。従って、非終端シンボルのTitle(タイトル)は終端シンボルのCHARの0個以上の発生から作られる。演算子“|”は選択できる表現を意味する。従って、上記プロダクションによればentity(要素)は、Text、TableEntity、Listの内のいずれかになる。また、幾つかのシステムは、属性を要素タイプへ付与する能力を有する。従って、例えば、Section(というシンボル)は、使用可能な言語の集合からの値を用いて、Section(というシンボル)に言語と称される属性を付与してもよく、特定なSection(というシンボル)は、該言語集合内の要素である可能性がある。

【0007】たいていのシステムは上述された方法で文書の論理構造を記述するための機能も提供する。次に、特定の文書は、例えば異なる要素同士の間の相互関係を記述するプロダクションに従うツリー構造等の階層的構造と対応する。図1を参照すると、汎用ツリー構造とルートノード、内部ノード及び子(終端としても知られている)を備えて示される。ルートノードはツリーの基本ノードである。ルートノードは、内部ノードとして図1 40に示されるその後の全てのノードの親である。子即ち終\*

\*端は、ツリーの最も低い要素ユニットであり、内部ノードの子孫である。このような階層的ツリー構造を用いて、文書のクラスの論理構造に記述される文法は、汎用論理構造と呼ばれる。次に、文書の具体例(instance)、即ち文法に従うツリーは特定の論理構造と呼ばれ、文書のクラスの具体例の記述である。例えば、特定のアーティクルは、全アーティクルのクラスの具体例である。形式構造(form structure)を記述する上記説明された文法とは異なる文法は、特定の論理構造と対応する特定の形式を備える汎用論理構造の別の例である。

【0008】バッチーオリエンテッドシステムの処理の最後に印刷される画像と同様に、編集中にスクリーン上に表示される画像は、文書の特定の論理構造から自動的に確立される。一般的にこのことは、汎用文書構造の各要素タイプが表示される又は印刷される方法を指定するプレゼンテーションルールに基づく。プレゼンテーションルールを指定する方法は一方のシステムからもう一方のシステムへ変更する。一般的に周知のシステムは、各文書要素タイプに付加される属性(特性)シート又は属性テーブル(attribute table)の概念(notion)を用いる。属性テーブルは、クラスの汎用構造において画定される要素タイプ毎に、各要素タイプの特性フォーマット属性の集合を含む。論理構造に関して上記記載されたアーティクルの例を用いて、要素タイプの幾つかに対してありうる属性テーブルは、以下で明らかにされる。疑問符は、値がユーザによって変更可能であり、疑問符(question mark)の後に続く値がデフォルト値(default value)である属性に対して用いられる。

【0009】

Article: FontFamily	=	? (Times)
MainTextFontShape	=	? (Roman)
MainTextFontSize	=	? (10)
MainTextLineSpacing	=	? (13)
MainHeadingFontShape	=	? (Bold)
HeadingFontShape	=	? (Italic)
TextAreaWidth	=	? (312)
TextAreaHeight	=	? (528)
AbsoluteTopMargin	5a	? (100)

5

6

```

LeftMargin          = ? (100)
RightMargin         = LeftMargin+TextAreaWidth
Language            = English
PAGINATE (AbsoluteTopMargin, TextAreaHeight)
Paragraph: ParagraphNumber = COUNTIN (Section)
Indentation         =
IF ParagraphNumber  = 0 THEN 0 ELSE em (FontSize)
JUSTIFY (LeftMargin, RightMargin, Indentation,
         Formatting, Mode, LineSpacing, Language)

```

【0010】上記例は、ア－ティクルとパラグラフの2つの論理文書要素タイプについてプレゼンテーション属性の値を備える属性テーブルを示す。上記テーブルにおいて、2つの手続き呼び出し(procedure call)がある。これらの手続き呼び出しは、ARTICLE (ア－ティクル)の場合、テーブル(表)の最後でPAGINATE (ページ付け)として示され、PARAGRAPH (パラグラフ)の場合、テーブルの最後でJUSTIFY (行揃え)として示される。これらの手順は、ア－ティクルをページに、パラグラフを行に、それぞれ分割する。(属性テーブルに設定され得る)これらの手順のパラメータは、これらの手順の結果を決定する。例えば、パラメータは、文書の画像、画像がページ中にレイアウトされる方法、及びパラグラフが行に分割される方法を決定する。

【0011】上記記載された従来技術の方法を用いることに多くの欠点がある。フォーマッティング手順はシステムの実現に隠れ、ユーザに接近しがたい。これらの手順のあらゆる変更は、システムの内部データ構造及び他のメカニズムについての詳細な知識を必要とする。ユーザが制御する唯一のことは、これらの手順のパラメータとして動作する属性である。また、結果を見ることができなければ、これらのパラメータの内の1つ以上の値を変更させることでフォーマッティングの結果を予測することは困難である。更に、大抵のシステムにおいて、プレゼンテーション規則に関する属性テーブルは文書に記述される論理構造の一部分を形成する。このことは、処理情報を論理構造情報と混合させるという欠点を有する。

#### 【0012】

【発明が解決しようとする課題】従って、必要とされるのは、論理構造の仕様を例えばレイアウト構造等の処理情報から分離させるための方法及び装置と、レイアウト構造を特定するための方法及び装置と、2つの異なる種類の構造(即ち論理構造とレイアウト構造)をコーディネート(調和)させるための方法及び装置と、レイアウト構造が論理構造とコーディネートするように、論理構造の一要素(member)を与えられたレイアウト構造の一要素を計算する(compute)ための方法及び装置と、である。

【0013】従って、本発明の目的は、文書の汎用論理構造の仕様が関係属性文法として表現されることを可能にする方法及び装置を提供することである。

10 【0014】本発明の別の目的は、論理的に構造化された文書のクラスのレイアウト構造を関係属性文法として特定するための方法及び装置を提供することである。

【0015】更に別の本発明の目的は、論理要素タイプとレイアウト要素タイプの間のコーディネーションを関係属性文法の点からコーディネーション文法として特定するための方法及び装置を提供することである。

20 【0016】本発明の別の目的は、潜在的な曖昧さを除くための、又は文書の論理構造から得られる対抗するレイアウトの中からどれかを選ぶための優先的な文(staement)を提供することである。

【0017】更に本発明の目的は、上述された構造(construction)を用いて、論理的に構造化された所与の文書のための最適レイアウトを計算することである。

#### 【0018】

【課題を解決するための手段と作用】これらの及び他の目的を実現し、先に説明された欠点を克服するために、請求項第1に記載の本発明は、構造化文書の汎用論理構造が与えられると、構造化文書の最適レイアウト構造を決定する、有限状態マシンで実現される方法であって、関係属性文法で前記構造化文書の前記汎用論理構造を指定し、前記関係属性が二項関係で指定されるステップと、関係属性文法で構造化文書のクラスのための汎用レイアウト構造を指定するステップと、前記構造化文書の前記汎用論理構造と前記構造化文書の前記汎用レイアウト構造のためにコーディネーション文法を決定し、関係属性文法で前記コーディネーション文法を表現するステップと、出力レイアウト構造中の潜在的な曖昧さを取り除くために優先仕様を決定するステップと、前記レイアウト構造を前記コーディネーション文法に基づいて前記汎用論理構造にリンクするステップと、前記優先仕様を用いてリンクされたレイアウト構造に前記論理構造を交換し、前記リンクステップ中に生成された曖昧さを取り除くステップと、を含む。

40 【0019】請求項第2に記載の本発明は、論理ツリー及び文法の集合に基づいて最適レイアウトツリーを決定する方法であって、論理構造情報が処理情報と分離される論理構造文法に基づいて論理ツリーを決定するステップと、複数のレイアウト構造を生成するために予め定められた文法に基づいて前記論理ツリーを構文解析するステップと、選択された属性の重み付け値計算に基づいて

50

7

前記複数のレイアウト構造を評価し、前記重み付け値が優先仕様に基づくステップと、前記優先仕様ベースの評価に基づいて最適レイアウトツリーを決定するステップと、を含む。

【0020】

【実施例】本発明は、好ましくは、有限状態マシン(finite state machine)ベースのコンピュータドキュメントハンドリングシステムで実現される。この型のシステムの一例は、プログラマブルでマイクロプロセッサベースの汎用コンピュータである。一方、本発明の実現は特定のコンピュータベースのドキュメントハンドリングシステムに限定される訳ではない。有限状態マシンは、好適な実施例の以下の詳細な説明より明白になる処理要求のために要求される。

【0021】文書は、周知のワードプロセッサパッケージを介してユーザによって汎用コンピュータで入力されることが好ましい。将来のアプリケーションにおいて、ワードを識別し、認識する能力を有するスキャナを使用可能である。

【0022】論理及びレイアウト構造の仕様は、一般的にユーザにはみえない。ユーザは所望される文書のタイプを指定し、コンピュータベースのドキュメントハンドリングシステムは、予め定められた構造のライブラリからの単数又は複数の構造を指定する。加えて、システムのエキスパートユーザのために、所望される構造がシステムライブラリにおいて使用できない場合、ユーザはユーザ自身の構造を指定できる。ここに記述されるように、次に、特定のレイアウト及び論理構造は得られるコーディネーション文法及び優先的な仕様を用いてコード\*

article → paragraph paragraph paragraph

このプロダクションは、アーティクルが3つのパラグラフから作られることを指定する。

【0026】図2を参照すると、上記記載された例示的な文法に属する又は文法に従うツリーの説明的な例が示される。全てのこのツリーのノードは、文法Gからのシンボルでラベル付けされる。親を備えないノードであるルートノードは、Gの開始シンボルとしても呼ばれるシンボルarticle (アーティクル) でラベル付けされる。子孫を有さないノードであるツリーの葉は、例えばWORD (ワード、単語) 等の終端シンボルでラベル付けされる。図2から容易に理解されるように、その子と共に全ノード (即ちその中間の子孫) は、幾つかのプロダクション体系の具体例を形成し、ノードがプロダクション体系の左側を表現し、子がプロダクション体系の右側を表現する。

【0027】これらのシンボルの各々は、本明細書の従来技術(Background)において前記記載されたように、従来技術と関連付される様々な属性を有することができる。例えば、非終端アーティクルは言語(language)と呼ばれる属性を有することができる。この属性の値は、ア

8

\*ィネートされ、リンクされ、最適化される。

【0023】文法は、構造化文書内の構造 (即ち、論理構造及びレイアウト構造) を指定するために用いられるルールである。文法は、文書のクラスの論理構造を指定する極めて自然な手段であり、ここでは、論じられる文書の構造を画定するために用いられる。詳細な記述は、発明の特徴を説明するために文書の単純なクラスを用いる。しかしながら、本発明が非常に複雑な構造にも同様に適することは理解される。従って、例えば、アーティクル等の文書の極めて単純なクラスの論理構造の例示的な文法は、以下の文法Gで説明できる。

【0024】

article → paragraph +  
paragraph → word +

これらの規則はプロダクションとも呼ばれる。文法Gは単に、現実的なアーティクルの構造の極めて単純な削り取られたバージョンを表現する例示的な文法にすぎない。簡潔さ及び説明的目的のために、アーティクルを説明する上記文法Gが用いられる。演算子“+”は、すぐ前のシンボルの1回以上の発生を意味する。文法Gは、2つの非終端シンボル (アーティクルとパラグラフ) 及び1つの終端シンボル (WORD) を有する。この文法は、アーティクルが1つ以上のパラグラフから作られ、次にパラグラフが1つ以上のワードから作られることを伝える。上記プロダクションはプロダクション体系 (production scheme) と呼ばれる。第1のプロダクションスキーマの具体例は、例えば、以下のように成り得る。

【0025】

アーティクルが書かれる言語を表す。パラグラフは、例えばアーティクル中のパラグラフ数を表す数と呼ばれる属性を有することができる。終端シンボルWORD (ワード) は、例えば内容等の属性を有することができ、その値はワードを形成するストリング(string)になる。既存のシステムはこれらの属性に値を与える単純な代入法を用いる。一方、本発明は属性、属性の値、及び属性間の従属性を指定するためにより論理的な表記法(logical notation)を用いる。例えば、既存のシステムにおいてアーティクルが英語であるという事実は、可変的な言語への代入として符号化されるだろう。

【0028】article : language = English

一方、本発明によれば、属性はシンボルと属性値の間の双方向関係になる。既存のシステムに関して上述された言語の属性は、本発明に従って次のように表現される。

【0029】language (article, English)

従って、言語の論理的述語(logical predicate) は、アーティクルと英語の間を満足させる。上記示された構文 (シンタックス(syntax)) 中の文(statement) は、論理的事実と呼ばれる。

【0030】既存のシステムとの別のバリエーションでは、本発明は属性の仕様を文法プロダクション自体と関連づける。このことにより、単一の文法プロダクション\*

number (paragraph [1], 1)

number (paragraph [i+1], x +1) if number (paragraph (i), x)

第1の事実は、最も左側のパラグラフの数が1になることを指定する。第2の論理規則は、i番目のパラグラフの数がxである場合、(i+1)番目のパラグラフの数はx+1であることを指定する。シンボル同士間の別の例示的な属性の従属性は、値がアーティクル中のパラグラフの数を表す属性カウントになりうる。このような属性の値は次の論理規則によって指定され得る。

【0031】

count (article, y) if number (paragraph [last], y)  
上記規則は、最も右側のパラグラフの子の数属性の値がyである場合、アーティクルのカウント属性の値がyになることを指定する。この例において、lastはあらゆる反復シーケンスの最後のアイテム(項)を示す特殊語である。従って、カウントは、そのカウントの値がアーティクルの子の属性の値から合成されるので、アーティクルの合成属性である。従って、例示的目的のために、上記識別された属性仕様と共に文法GになるG<sub>log</sub>を示す。

【0032】図3は、G<sub>log</sub>に属す属性ツリーを示す。このツリーは図2のツリーと構文的に(即ち、構造的に)等しい。しかしながら、図3のツリーは上述された属性値で飾られている。これらの値は、G<sub>log</sub>のプロダクションに付加される属性仕様を満足させる。これらの属性文法は関係があり、属性仕様は論理的事実及び規則によって与えられる。

【0033】汎用レイアウト構造は、論理構造に関して上述された方法と同じ方法で定義されることもできる。例えば、非常に単純な例示的な文法G'はページシーケンスとして文書のレイアウトを記述し、各ページは行のシーケンスである。

TextAreaWidth (page, w) if TextAreaWidth (articleLayout, w)

TextAreaWidth (line, w) if TextAreaWidth (page, w)

他の継承属性は、文法の開始シンボルから他のシンボルへ同様に下りていく。この例においてG<sub>lay</sub>は上述された属性仕様を備える文法G'を示す。図4は、文法G<sub>lay</sub>に属すツリーを示す。

【0036】一度、G<sub>log</sub>及びG<sub>lay</sub>のツリー構造が決定されると、論理構造はレイアウト構造とリンク(関係)されることが必須である。このリンクは、汎用構造のレベルで行われる。リンクを行えば、コーディネーション文法は関連する論理及びレイアウト構造に対して得られる。従って、次の制約条件を満たさなくてはならない。まず、articleLayout(アーティクルレイアウト)でラベル付けされたルートノードを備えるレイアウト構造ツリーになるべきであり、これはアーティクルでラベル

\*中の異なるシンボルの属性値間の従属性の仕様が可能になる。以下の例示的な論理的事実において：

※【0034】

1. articleLayout → page +

2. page → line +

3. line → UNIT +

G'中の非終端シンボルはarticleLayout(アーティクルレイアウト)、page(ページ)、及びline(行)である。ArticleLayout(アーティクルレイアウト)は、G'の開始シンボルである。UNIT(ユニット)は、終端シンボルである。

【0035】次に、文法G'は幾つかの例示的な属性を与えられ得る。例えば、終端シンボルUNITは、内容(content)と呼ばれる属性を有し、その値が、そのUNITによって保持される実際の内容であるテキストストリングになる。別の属性は不良(badness)と呼ばれる属性になることができ、特定の行上の全てのUNITに合わせる妥協を表す。例えば、UNITの内容は特定の行上の全てのUNITを収容するために伸ばされる又は縮められなければならないことがある。不良の仕様の詳細は、例示的なものであり、従来技術においてかなり標準的である。行と関連付けられる幾つかの他の例示的な属性は、MainTextLineSpacing(メインテキスト行間隔)、MainTextFontSize(メインテキストフォントサイズ)、LeftMargin(左マージン)、TextAreaWidth(テキスト領域幅)等がある。このような属性は一般的に完成レイアウト構造のために指定され、即ち、アーティクルレイアウトのレベル(段階)で指定され、例えばページや行等の様々な部分構造は、その構造の親からこれらの属性を単純に継承する。従って、一方の属性は、上記第1及び第2のプロダクションにそれぞれ付加される次の属性規則を有することができる：

付けされたルートノードを備える各論理構造ツリーに対応する。次に、行でラベル付けされるルートノードを備えるレイアウトツリーのシーケンスになるべきであり、これはパラグラフでラベル付けされたルートノードを有する各論理構造部分ツリーと対応し、パラグラフが行のシーケンスとしてレイアウトされるという事実を示す。次に、論理構造中の全WORDは、レイアウト構造中のUNITに対応する。上述されたようにこのようなコーディネーション(又はリンクージ)も、文法とし表現できる。上述されたコーディネーションの場合、それぞれが単一のプロダクションを有する以下の文法が使用可能である。

【0037】

G<sub>c1</sub>: article → articleLayout



G<sub>c2</sub>: paragraph → Line+

G<sub>c3</sub>: WORD → UNIT

G<sub>c1</sub>、G<sub>c2</sub>、G<sub>c3</sub>は、論理要素をレイアウト要素とリンクするコーディネーション文法である。また、コーディネーション文法は論理側からレイアウト側へ情報を渡すように非常に自然な位置にある。従って、以下の属性仕様をG<sub>c3</sub>に取り付けることによって、

content (UNIT, c) if content (WORD, c)

となる。全WORDの内容は、UNITの内容がコーディネートされるレイアウト側に転送される。

【0038】図5はG<sub>log</sub>に従う例示的な論理ツリーT<sub>log</sub>と、G<sub>lay</sub>に従うレイアウトツリーT<sub>lay</sub>と、論理要素ノードをレイアウト要素ノードにリンクするコーディネーションツリーを示す。以下の順序属性は、コーディネーションによって満足される。

1. パラグラフP<sub>1</sub>が、論理構造中のP<sub>2</sub>の左側に現れると、次にそのレイアウト画像(シーケンスL<sub>1</sub>, L<sub>2</sub>, L<sub>3</sub>)はP<sub>2</sub>のレイアウト画像(シーケンスL<sub>4</sub>, L<sub>5</sub>)の左側に現れる。このように、左-右順序付けはコーディネーションによって保持される。

2. 先祖の順序付けもコーディネーションによって保持される、即ち、(アーティクルレイアウトである)アーティクルのレイアウト画像は行であるパラグラフのレイ\*

$\text{cost}(\text{paragraph}, c) \text{ if } \bigwedge_i \text{badness}(\text{line}[i]b_i) \wedge \sum_i b_i = c$

【0042】上記属性規則は、パラグラフのコストがパラグラフ中の全行の不良値の総計であることを指定する。容易にわかるように、上記の識別された属性を用いてパラグラフのレイアウトをハンドルする多くの方法がある。例えば、ユーザは最低コスト属性を有するパラグラフを選ぶことができる。略式に、このような優先情報は、以下の文(statement)によって G<sub>c2</sub>に関連付けられる。

【0043】PREFERENCE STATEMENT: MIN (cost)

【0044】この仕様は、コストの最小値を有するパラグラフのレイアウトを選択することを記述した。属性規則及び優先制約条件と共にG<sub>c2</sub>は、G' <sub>c2</sub>と呼ばれる。優先仕様は、例えばコスト等の選択的基準に対してどのレイアウトがより良いかを決定できる根拠(basis)を提供する。コストは使用可能な多くの優先制約条件の単に1つにすぎず、コストは説明の目的のために単に用いられていることに留意されたい。優先文を用いて、ユーザはまた優先文に与えられた属性に基づいて部分最適レイアウトを拒否できる。

【0045】上記の例に示されるように、文書の論理構造は、属性文法を用いて表現可能である。このような文法は文書のクラスを記述し、各文書の具体例はその文法のツリーに対応する。意味情報(semantic information)も、属性を用いて表現可能である。属性仕様は事実と規則の形式で、文法のプロダクションに付加可能である。

\*アウト画像より上にある。

【0039】論理的なレイアウト及びコーディネーション文法の特徴は、レイアウトの仕様が極めて曖昧であるということである。パラグラフが与えられると、仕様はそのパラグラフを多くの異なる方法で行にレイアウトできる。図6は図5と同様に論理ツリーT<sub>log</sub>を示す。一方、図6の論理ツリーは違う方法でレイアウトされた、即ち、異なるレイアウトツリーT<sub>lay</sub>にリンクされた。汎用構造(G<sub>log</sub>, G<sub>lay</sub>, G<sub>c1</sub>, G<sub>c2</sub>, G<sub>c3</sub>)は、同じままであることに留意されたい。本発明によれば、これらの曖昧さは、非常に新規性のある固有の方法でハンドル(処理)される。上述されたように、パラグラフは、G<sub>c2</sub>を用いて単一プロダクションと調和された。

【0040】G<sub>c2</sub>: paragraph → Line+

加えて、属性不良はこれらの行と関連付けられた。属性コスト(費用)もこれらのパラグラフと関連付けられてもよい。例えば、この属性の値は、パラグラフを作成するために全行の不良の総計を出すことによって得られる。従って、このコスト属性は、以下の規則をG<sub>c2</sub>と関連付けることによって、指定できる。

【0041】

【外1】

このような文法は、属性が二項関係として指定されるので、関係属性文法(relational attribute grammar)と呼ばれる。この種類の属性仕様は、属性仕様が値を変数に代入するために基本的に用いられる従来技術に記載された属性仕様と対比されるべきである。

【0046】加えて、処理情報は好ましくは論理構造の仕様から完全に分離されるべきである。レイアウト構造は関係属性文法として指定されることもできる。論理文法とレイアウト文法間のリンケージ(linkage)又はコーディネーションも関係属性文法として表現可能である。仕様におけるあらゆる曖昧性は、ある属性が選択基準(本明細書の例ではコスト)である場合、最低値でレイアウトを選択するために優先仕様を用いることによって、解決される。仕様技術は論理構造の単純な例を用いて説明されているが、これらの原理は、このような関係属性文法を用いて記述され得る全構造に対して適用可能である。

【0047】上記記述の場合、文書をレイアウトすること(又はレイアウト処理)の問題点は、次のように減らされる。論理構造文法G<sub>1</sub>、レイアウト構造文法G<sub>2</sub>、及びそれらの文法に対するコーディネーション文法G<sub>c1</sub>... G<sub>ck</sub>が与えられ、論理文書の具体例(即ち、特定の論理構造又は論理ツリー)T<sub>log</sub>が与えられると、レイアウトツリーT<sub>lay</sub>は必ず決定されなければならない。T<sub>lay</sub>はG<sub>2</sub>に属し、コーディネーション文法によ

13

って与えられるコーディネーションに従って $T_{log}$ と必ずコーディネートされなければならない、多くの異なるこのようなレイアウトツリーの中で、最適でなければならない、即ち、例えば、全ての選択される優先属性に対して最小値を有さなければならない。

【0048】 $T_{lay}$ を上記規則に適應させる処理は、文法に従ってストリングを構文解析(parsing)するための既存のアルゴリズム及び技術を用いる。構文解析は、当業者に周知であり、下記に簡略に記載される。簡略的に言うと、ストリングは、文法のシンボル(終端シンボルと非終端シンボル)の線形シーケンスである。文法G及びストリングSが与えられると、構文解析の処理は、このツリーのルートノードがGの開始シンボルでラベル付けされ、ツリーの葉がストリングSの要素でラベル付けされ、ツリーの内部ノードが文法のプロダクションに従う様なSに対してツリー構造を構成する処理として定義される。シンボルSのストリング及び文脈自由文法(context free grammar)G(参照された全文法が文脈自由文法であったことに留意されたい)が与えられる場合、このようなツリーが存在すれば、Sに対して解析ツリーを出力する周知のアルゴリズムがある。このようなツリーが存在しなければ、これらのアルゴリズムは故障メッセージ(failure message)を出力する。文法Gのプロダクションに付加される属性要求仕様が与えられると、周知のアルゴリズムが存在し、解析ツリーTを与えられ、文法の属性仕様及び従属性規則を満足させるような属性値でそれを装飾できる。これらのアルゴリズムは、評価者(evaluator)と呼ばれる。このような構文解析系及び属性の評価者がわかれば、レイアウトツリーの構成は、底から上にステップ状にすすむ。図7、8、9及び10は、所与の論理ツリーのレイアウトツリーの構成を段階(ステージ)的に説明する。それらはコーディネーション文法と同じだけ多くのステージがある(前記例においては3ステージ)。図7は、例示的なアーティクルの論理構造ツリーを示す。

【0049】図8に示されるステップ1は、(論理構造の最低レベルにある)WORDからUNITのストリングを生成するために用いられる。UNITは我々に明示的には与えられない。与えられるのは、論理構造ツリー(図7参照)である。最低レベルの論理シンボルのコーディネーション文法を用いて、WORDと同じだけ多くのUNITが生成される。次に評価者は、各UNITの内容属性を計算するために呼び出される。

【0050】図9に示されるステップ2において、構文解析系は、パラグラフと同じだけの回数、呼び出される。このような呼び出し毎に、構文解析系で用いられる文法( $G_{II}$ )は2つのプロダクションを有する。第1のプロダクションは $G'_{c2}$ である。第2のプロダクションは、 $line \rightarrow UNIT+$ である。構文解析系に入力されるストリングは、前の段階より生成されたUNITのストリン

14

グである(図8参照)。従って、第1パラグラフ $P_1$ の構文解析系呼び出しの場合、 $P_1$ に渡されるストリングは $P_1$ 中のWORD全てについてステージIで生成されるUNITのシーケンスである。ステージII(図9)は、レイアウト構造中に行を確立する。文法 $G_{II}$ は曖昧なので1つを越えるこのような行のシーケンスは生成されるだろう。選択をするために、評価者は呼び出され、全属性の値を計算する。この例において、コストの値は可能な構文解析毎(即ち、パラグラフを行に分解する方法毎)に計算される。コストの最低値(又は特定された優先制約条件の選択された”最良”値)を有する構文解析が選択される。この処理は全パラグラフに対して実行される。

【0051】第3ステージでは、構文解析は、論理側のアーティクルシンボルに対して行われる。このステージのために用いられる文法は、以下の $G_{lay}$ のプロダクションと同様にプロダクション $G_{c1}$ からなる $G_{III}$ である。

【0052】

articleLayout  $\rightarrow$  page+  
page  $\rightarrow$  line+

この文法のための構文解析系への入力ストリングは、ステージIIから生成された行のシーケンスである。

【0053】上記記載されたことは、論理、レイアウト、及びコーディネーション構造の文法的な記述を与えられた所与の論理ツリーに対する最適レイアウトツリーを築くためのステージ処理である。このタイプの最適化は成層最適化(stratified optimization)と呼ばれる。従って、あらゆるステージの出力は、次のステージが呼び出される前に固定される。例えば、ページにも付加されるコストの概念にもなり得る。このようなコストは、ページがどの様にレイアウトされるか、パラグラフがどの様にページを分割するか等を考慮に入れることができる。説明された手順は、パラグラフの行への最適分割を最初に得るだろう。次に、これらの行は、ページ中の行の最適分割を行う構文解析系に入力される。このように最適化は層状構造において実行される。

【0054】なお、図中のARTICLEは、アーティクル(論文)、PARAGRAPHはパラグラフ、WORDはワード(語)、ARTICLE COUNT:2はアーティクル数:2、PARAGRAPH NUMBER:1はパラグラフ数:1、PARAGRAPH NUMBER:2はパラグラフ数:2、WORD CONTENTはワード(語)内容、ARTICLE LAYOUTはアーティクルレイアウト、PAGEはページ、LINE BADNESSは行不良、UNIT CONTENTはユニット(単位)内容、lineは行、をそれぞれ示す。

【0055】

【発明の効果】本発明は、関係属性文法の間で関係属性文法及びコーディネーション文法を用いて構造化文書のレイアウトを指定するための方法及び装置を提供する。加えて、本発明は論理構造及びコーディネーション文法から得られるレイアウト構造中の潜在的な曖昧さを除く

15

ために優先的な制約条件を指定し、部分的に最適化された解を除去するためにこれらの優先的な制約条件を用いるための方法及び装置を提供する。

【図面の簡単な説明】

【図 1】 例示的なツリー構造を示す線図である。

【図 2】 アーティクルを記述する文法に属するツリーを示す。

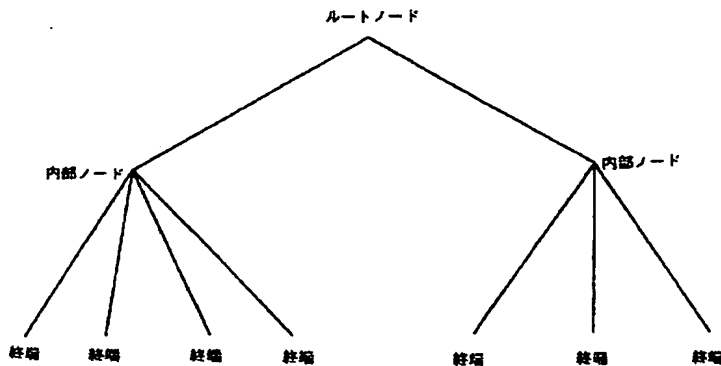
【図 3】 例示的な汎用論理構造を表現する属性ツリーを示す。

【図 4】 例示的な汎用レイアウト構造のための属性ツリーを示す。

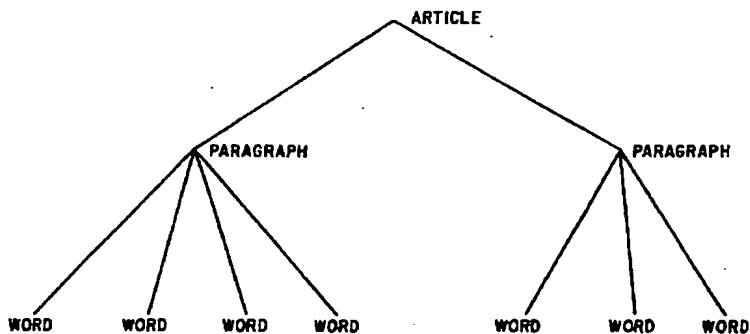
【図 5】 例示的な文法の集合に従う論理的なレイアウト及びコーディネーションツリーを示す。

【図 6】 コーディネーションツリーに於ける本質的な曖昧\*

【図 1】



【図 2】



16

\* 味さを強調する例示的なツリー構造を示す。

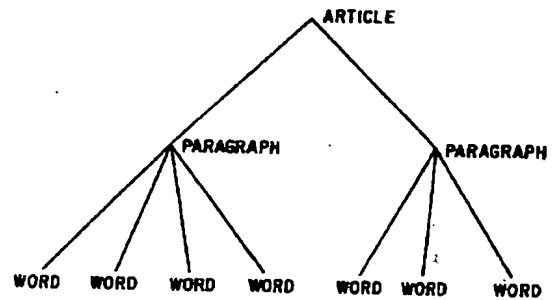
【図 7】 本発明に従って、論理ツリーを与えられる最適レイアウトツリーの計算 (computation) の初期ステージの表現を示す。

【図 8】 本発明に従って、論理ツリーを与えられる最適レイアウトツリーの計算の第 1 の中間ステージの表現を示す。

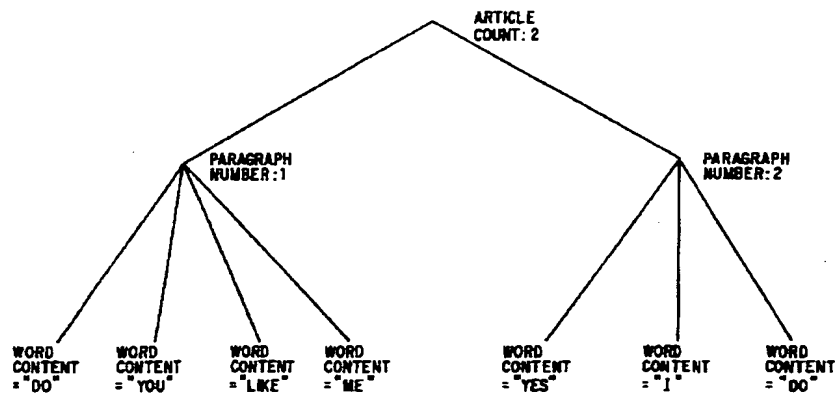
【図 9】 本発明に従って、論理ツリーを与えられる最適レイアウトツリーの計算の第 2 の中間ステージの表現を示す。

【図 10】 本発明に従って、論理ツリーを与えられる最適レイアウトツリーの計算の最終ステージの表現を示す。

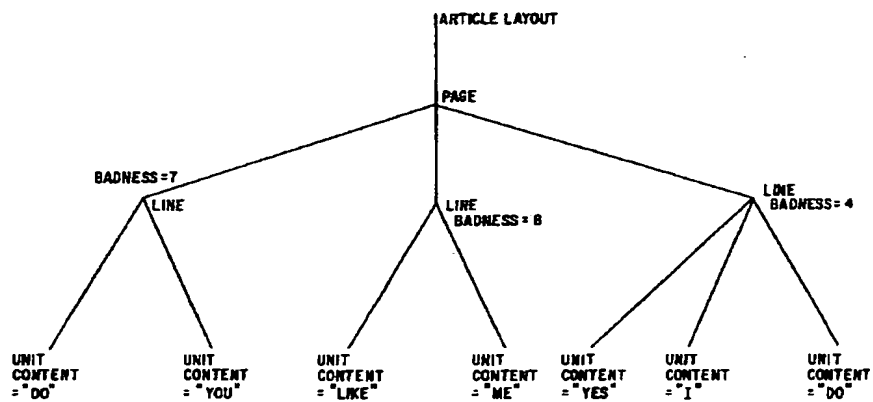
【図 7】



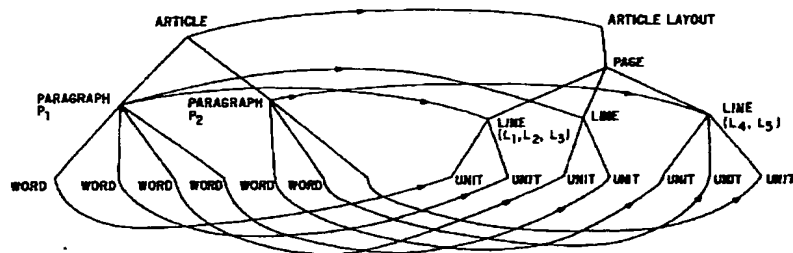
【図 3】



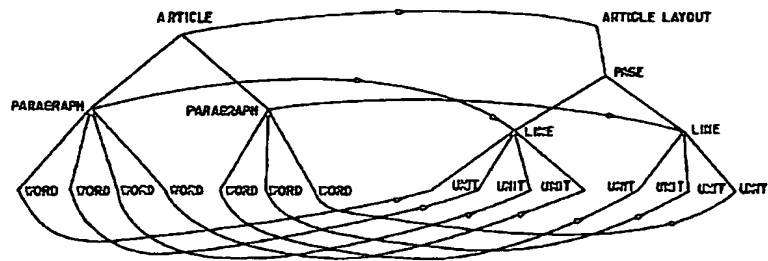
【図 4】



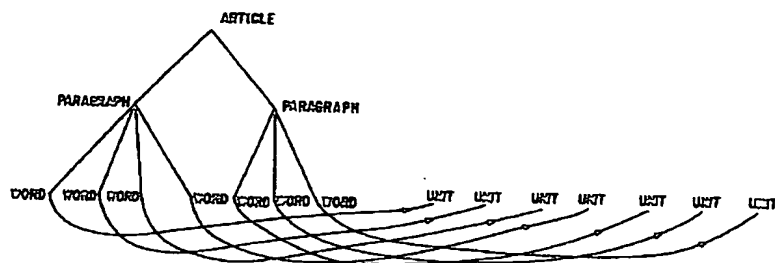
【図 5】



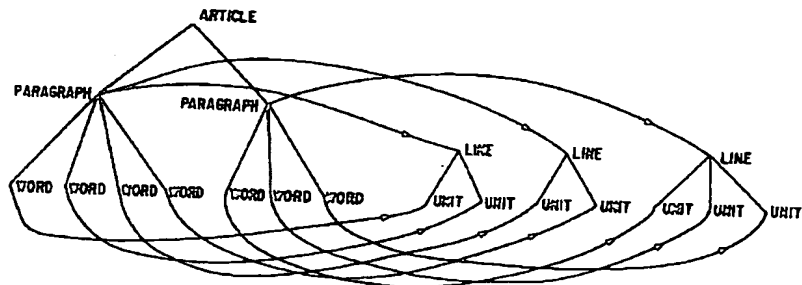
【図 6】



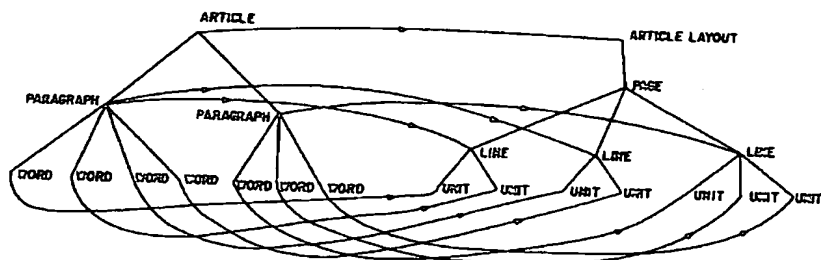
【図 8】



【図 9】



【図 10】



フロントページの続き

(72)発明者 アレン エル. ブラウン, ジュニア  
アメリカ合衆国 カリフォルニア州  
92075 ソラナ ビーチ グレンモント  
ドライヴ 425

(72)発明者 トシロー ワカヤマ  
アメリカ合衆国 ニューヨーク州 14580  
ウェブスター サウスウィック ドライ  
ヴ 14